CLAIMS

1. A method for identifying defective cells in a memory array, the method comprising steps of:

receiving a memory test command;

initiating a memory test in response to the memory test command, the memory test for determining addresses of defective cells in a random-access memory;

concluding the memory test; and

compressing the addresses of defective cells to provide compressed address data.

2. A method as claimed in claim 1 wherein the step of initiating a memory test includes a step of storing addresses of defective cells in a temporary memory array.

3. A method as claimed in claim 1, further including a step of storing the compressed address data in a map memory array.

4. A method for accessing a memory array, the method comprising steps of:

requesting an address for one or more cells comprising a first memory array;

analyzing the address to determine when the address matches an uncompressed address in a temporary memory array, and, when the address does not match any uncompressed address stored in the temporary memory array, the method includes steps of:

analyzing the address to determine which portion of compressed data stored in a third memory array containing compressed addresses of defective cells in the first memory array to decompress;

decompressing the portion of compressed data to provide expanded data;

writing the expanded data to the temporary memory array;

comparing the expanded data to the address to determine when the address corresponds to an expanded datum of the expanded data; and

routing the address to a second memory array when the address and a datum from the decompressed data match.

5. A method as claimed in claim 4, further including a step of routing the address to the first memory array when the comparing step determines that the address does not match any datum from the decompressed or compressed data.

6. A method as claimed in claim 4, wherein the step of routing the address to a second memory array includes a step of reading data via the address.

7. A method as claimed in claim 4, wherein the step of routing the address to a second memory array includes a step of writing data via the address.

8. A method as claimed in claim 4 wherein the step of routing the address to a second memory array comprises a step of routing the address to a separate portion of the first memory array when the address and a datum from the decompressed data match.

9. A method for accessing a memory array, the method comprising steps of:

receiving a memory array access request including a requested address;

generating a first hash code from the requested address;

comparing the first hash code to hash codes for decompressed addresses stored in a temporary memory array;

determining if an address stored in the temporary array corresponds to the requested address when a match is found between a hash code for a decompressed address and the first hash code; and

routing the memory array access request to a spare memory array when an address stored in the temporary array corresponds to the requested address.

10. A method as claimed in claim 9, wherein, when the comparing step indicates that no match is found between the first hash code and the hash codes for decompressed addresses stored in the temporary memory array, the method includes steps of:

determining which portion of a map memory array to decompress, the map memory array storing compressed addresses of defective memory array cells in a first memory array;

decompressing compressed data from the portion to provide decompressed addresses; and

storing the decompressed address in the temporary memory array.

11. A method as claimed in claim 10, wherein the step of decompressing compressed data includes a step of generating a hash code for each decompressed address, and wherein the method further comprises a step of comparing the first hash code to the hash codes for the decompressed addresses stored in the temporary memory array.

12. A method as claimed in claim 11 wherein, when a match is found between a hash code for a decompressed address and the first hash code, the method includes a step of routing the memory array address request to the spare memory array.

13. A method as claimed in claim 12, further comprising a step of sending a wait command prior to said step of determining which portion of a map memory array to decompress.

14. A method as claimed in claim 13, further comprising a step of sending a cancel wait command prior to said step of routing the memory array address request to the spare memory array.

15. A method as claimed in claim 12 wherein, when no match is found between the first hash code and the hash codes for the decompressed addresses stored in the temporary memory array, the method includes steps of:

sending a cancel wait command; and

routing the memory array access request to the first memory array.

16. A method as claimed in claim 15 wherein:

the step of receiving a memory array access request includes a step of receiving a memory array access request including a requested address from a processor;

the step of sending a wait command includes a step of sending a wait command to the processor; and

the step of sending a cancel wait command includes a step of sending a cancel wait command to the processor.

17. A method for identifying defective cells in a memory array, the method comprising steps of:

receiving a request for accessing an address;

analyzing the address to determine when the address matches an address stored in a temporary memory array, and, when the address does not match any address stored in the temporary memory array, performing steps of:

analyzing the address to determine which portion of compressed data stored in a map memory array containing compressed addresses of defective cells in a first memory array to decompress;

decompressing the portion of compressed data to provide expanded data;

writing the expanded data to the temporary memory array; and

comparing the expanded data to the address to determine when the address corresponds to an expanded datum of the expanded data.

18. A method as claimed in claim 17, further comprising a step of routing the address to a first memory array when the address does not match any address stored in the temporary memory array or to an expanded datum of the expanded data.

19. A method as claimed in claim 17, further comprising a step of routing the address to a second memory array when the step of analyzing the address determines that the address matches an address stored in the temporary memory array.

20. A method as claimed in claim 17, further comprising a step of routing the address to a second memory array when the step of comparing the expanded data to the address determines that the address corresponds to an expanded datum of the expanded data or to an address stored in the temporary memory array.

21. A memory control circuit comprising:

storage control unit means coupled to a bus, the storage control unit means for accessing memory array units to retrieve data from a first memory array unit in response to memory array access requests delivered via the bus;

first memory array means coupled to the storage control unit means, the first memory array means for storing data;

second memory array means coupled to the storage control unit means, the second memory array means for replacing cells determined to be defective in the first memory array means; and

means for compressing data describing memory array addresses corresponding to cells determined to be defective in the first memory array means to provide compressed addresses and for decompressing compressed addresses to

provide decompressed addresses, the compressing means coupled to the storage control unit means.

22. A memory control circuit as claimed in claim 21, further comprising third memory array means for storing the compressed addresses, the third memory array means coupled to the compressing means.

23. A memory control circuit as claimed in claim 22, further comprising fourth memory array means coupled to the storage control unit means, the fourth memory array means for temporarily storing the decompressed addresses from the compressing means.

24. A memory control circuit as claimed in claim 23 wherein the first, second, third and fourth memory array means comprise random-access memories.

25. A memory control circuit as claimed in claim 23 wherein the compressing means is for:

identifying a starting and an ending address for a group of adjacent defective cell sites in the first memory array means;

reconfiguring the addresses of the group as the starting address and a difference between the starting address and the ending address to provide compressed addresses; and

supplying the compressed addresses to the third memory array means.

26. A memory control circuit as claimed in claim 25 wherein the compressing means is further for:

accepting addresses of a group of adjacent addresses describing defective cell sites in the first memory array means as a starting address and a difference between the starting address and the ending address; and

reconstructing individual addresses of the defective cell sites to provide decompressed addresses.

27. A memory control circuit as claimed in claim 25 wherein the compressing means is additionally for writing the decompressed addresses to the fourth memory array means.

28. A memory control circuit as claimed in claim 21 wherein the storage control unit means, the first memory array means, the second memory array means, the compressing means and the third memory array means comprise an integrated circuit.

29. A memory control circuit comprising:

a storage control unit coupled to a bus, the storage control unit for accessing memory array units to retrieve data from a first memory array unit in response to memory array access requests delivered via the bus;

a first memory array coupled to the storage control unit, the first memory array for storing data;

a second memory array coupled to the storage control unit, the second memory array for replacing cells determined to be defective in the first memory array; and

a data compressor that compresses data describing memory array addresses corresponding to cells determined to be defective in the first memory array to provide compressed addresses and that decompresses compressed addresses to provide decompressed addresses, the data compressor coupled to the storage control unit.

30. A memory control circuit as claimed in claim 29, further comprising a third memory array for storing the compressed addresses, the third memory array coupled to the data compressor.

31. A memory control circuit as claimed in claim 30, further comprising a fourth memory array coupled to the storage control unit, the fourth memory array for temporarily storing the decompressed addresses from the data compressor.

32. A memory control circuit as claimed in claim 31 wherein the first, second, third and fourth memory arrays comprise random-access memories.

33. A memory control circuit as claimed in claim 31 wherein the data compressor identifies a starting and an ending address for a group of adjacent defective cell sites in the first memory array, reconfigures the addresses of the group as the starting address and a difference between the starting address and the ending address to provide compressed addresses and supplies the compressed addresses to the third memory array.

34. A memory control circuit as claimed in claim 33 wherein the data compressor further accepts addresses of a group of adjacent addresses describing defective cell sites in the first memory array as a starting address and a difference between the starting address and the ending address and reconstructs individual addresses of the defective cell sites to provide decompressed addresses.

35. A memory control circuit as claimed in claim 33 wherein the compressing means is additionally for writing the decompressed addresses to the fourth memory array.

36. A memory control circuit as claimed in claim 28, wherein the storage control unit, the first memory array, the second memory array, the data compressor and the third memory array comprise an integrated circuit.

37. A computer comprising:

a processor;

a read-only memory storing instructions for operation of the processor;

a random-access memory array storing data;

a spare random-access memory array storing data corresponding to defective locations in the random-access memory array;

a storage control unit coupled to the processor, the read-only memory, the random-access memory and the spare random-access memory, the storage control unit accessing the read-only memory, the random-access memory and the spare random-access memory to retrieve data in response to commands from the processor; and

a data compressor coupled to the storage control unit, the data compressor compressing data indicative of addresses of defective storage locations in the random access memory array to provide compressed addresses, and decompressing the compressed addresses.

38. A computer as claimed in claim 37 wherein the data compressor comprises the processor executing instructions stored in the read-only memory.

39. A computer as claimed in claim 37 wherein the data compressor comprises circuitry incorporated within the storage control unit.

40. A computer as claimed in claim 37 wherein the data compressor comprises circuitry incorporated within the random-access memory array.